ESIEE PARIS

Curriculum 1st and 2nd year (E4-E5) 2020-2022

Major: International Master of Computer Science for Intelligent Systems

Technical Courses				
Title	**code**	**hours**	**credits**	**Period**
Networking	IMC-4103A	30	2,5	E4 Semester 1
Algorithm design	INF-4102B	30	2,5	
Computer architecture	IMC-4101C	30	2,5	
Signal and systems (elective)	OUAP-4109	30	2,5	
Programming tools (refresh)	PIM-4104	30	2,5	
Introduction to image analysis (elective)	OUAP-4215	30	2,5	
Real-time systems	IMC-4202B	30	2,5	
Graph and algorithms	IMC-4202C	30	2,5	
Introduction to robotics (elective)	OUAP-4213	30	2,5	
Statistics with Python (refresh)	PIM-4205	30	2,5	
Wireless networks	IMC-4302A	30	2,5	E4 Semester 2
Optimization algorithms	INF-4302B	30	2,5	
Machine learning 1	IMC-4302C	30	2,5	
Model checking (elective)	OUAP-4307	30	2,5	
Embedded operating systems	IMC-5102A	30	2,5	E5 Semester 1
Machine learning 2	IMC-5103B	30	2,5	
Image analysis	IMC-5101C	30	2,5	
Safety and security	IS-5111E	30	2,5	
Critical application development	IMC-5203C	30	2,5	
Data science for developers	IMC-5203A	30	2,5	
Computer Vision	IMC-5204B	30	2,5	
Distributed control systems	IS-5206E	30	2,5	

Course Content 1st and 2nd year (E4-E5)

Major: International Master of Computer Science for Intelligent Systems

Networking (Lynda Zitoune) (E4 semester 1)
Content: OSI model, layer abstraction; Paquet switching (OSI layer 2); Routing (CISCO routers, OSI layer 3); WAN (Wide Area Network) and VPN (Virtual Private Network). **Aim:** This course provides attendees with the most essential concepts from low to mid-level networking. The most widespread networking technologies are first introduced. It is then shown how these different technologies may be abstracted in the first layers of the OSI model. Routing is considered, with concrete example given on CISCO routers. Large networks, and their administration, are explained at last. **Bibliography** 1. Tanenbaum : Computer Networks 2. W. R. Stevens : TCP/IP Illustrated, protocols

Algorithm design (Nabil Mustafa) (E4 semester 1)
Content: Dynamic programming; Divide and conquer; Greedy algorithms; Introduction to NP-completeness and approximation algorithms **Aim:** We first recall basic notions on complexity of algorithms. We then present resolution methods from three widespread families of algorithms (divide and conquer, dynamic programming, greedy algorithms), all providing exact solutions to well identified problems. NP-completeness problems are presented at last, along with classical reduction proofs. Students should be able to diagnose whether a problem is NP-hard or not, and if it is, propose an approximate algorithm and bound its efficiency. **Bibliography** S. Dasgupta, C. Papadimitriou, U. Vazirani, Algorithms http://www.cs.berkeley.edu/~vazirani/algorithms/

Computer Architecture (Eva Dokladalova) (E4 semester 1)
Content: History of computing technology, the different components of a computer; communication between computers and the outside world; performance measurements; pipelining; memory management, project: Programming of a real-life application, i.e. autonomous robot navigation, based on the application of the acquired knowledge **Aim:** The purpose of this course is to cultivate an understanding of modern computing technology through an in-depth study of the interface between hardware and software. In this course, you will study the history of modern computing technology before learning about modern computer architecture and a number of its important features, including instruction sets, processor arithmetic and control, the Von Neumann architecture, pipelining, memory management, storage, and other input/output topics. **Bibliography** John L. Hennessy and David A. Patterson. 2011. Computer Architecture, Fifth Edition: A Quantitative Approach (5th ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Signal and Systems (Arben Cela) (E4 semester 1)

Content: Fourier and Laplace transforms; Theory of convolution; Sampling and reconstruction; Dynamical Systems; Integration methods (Euler, Runge -Kutta , Adams) ; Stability ; Controllability, Observability.

Aim: Introduction to necessary mathematical tools for analysis of signal and dynamic systems. Different concepts such as stability, controllability and observability are introduced through different well known application examples. Mastering of the related Matlab/Simulink toolboxes, introduced through different applications, is another objectives of this course. The course also aims at introducing the basics of continuous-time and discrete-time signals and systems, such as impulse and frequency responses, the Fourier Transform, Fourier Series, Discrete Fourier Transform, Finite Fourier Transform, as well as the Z-Transform and the Function Transfer of a system. The course also involves, as part of it, the study and design of digital filters.

Bibliography

1. K.J Aström and R. Murray, "Feedback Systems, An Introduction for Scientists and Engineers" Princeton University Press (http ://www.cds.caltech.edu/ murray/books/AM05/pdf/am08-complete 22Feb09.pdf)
2. Alan V. Oppenheim, Alan V. Oppenheim, Ronald W. Schafer, John R. Buck, Discrete Time Signal Processing, PRENTICE HALL (1999)

Introduction to Image Analysis (Jean Cousty) (E4 semester 1)

Content: Non-linear signal processing; Erosion, dilation, closing, opening; Skeletons; Watershed segmentation, connected operators.

Aim: The aim of this course is to provide the fundamentals of mathematical morphology. We introduce new concepts in non-linear signal analysis, then explain the basic operators used in mathematical morphology and their main properties, and skeletonization. The problem of image segmentation is then considered, with the very popular watershed segmentation approach. Non-linear filtering and detection are illustrated on a wide variety of problems.

Bibliography https ://perso.esiee.fr/ coustyj/EnglishMorphoGraph/

Real-Time Systems (Yasmina Abdeddaïm) (E4 semester 1)

Content: Real time scheduling algorithms; Feasibility analysis; Optimality analysis; Resource sharing; Real-time Linux (RTAI).

Aim: This course is an introduction to scheduling for hard real-time systems. We introduce the task model, the classical scheduling algorithms, and feasibility analysis based on this model. Scheduling algorithms will be tested in practice on a real-time kernel.

Bibliography

1. G. C. Buttazzo Hard Real-time Computing Systems : Predictable Scheduling Algorithms And Applications (Real-Time Systems Series) Springer-Verlag TELOS Santa Clara, CA, USA,2004
2. rtai : https //www.rtai.org/
3. freeRTOS : www.freertos.org

Graph and Algorithms (Jean Cousty) (E4 semester 1)

Content: Graph traversal; Connected components; Shortest path; Minimum spanning tree; Maximal flow.

Aim: This course is an introduction to the most popular algorithms produced by graph theory, and used in pattern recognition, combinatorics, AI, and problem resolution amongst others. It aims to provide attendees with the ability to: formalize a given problem in terms of graphs; identify whether the problem has a known solution or not; and in case not, suggest a new algorithm and evaluate its complexity.

Introduction to Robotics (Ting Wang) (E4 semester 1)

Content: Motion planning; Kinematics, Inverse kinematics; Sensors and actuators.

Aim: Understanding the structure of robotic systems and the modelling and simulation of robotic systems in MATLAB/Simulink. This course will also expose students to some of the contemporary happenings in robotics

Bibliography Industrial Robotics: Theory, Modelling and Control, Sam Cubero

Statistics with Python (Arben Cela) (E4 semester 1)

Content: An introduction to statistics using Python; basic notions of Python: datatypes; including vectors and arrays; indexing; functions writing; plotting data; visualization of data; Basic notions of statistical populations; samples, distributions, expected value, variance, co-variance, sensitivity, specificity, tests of means of numerical data, confidence intervals, regression.

Aim: At the end of this course, the students will be capable of using Python in order to conduct basic statistical analysis: they will be able to use input data (text file, Excel file, Matlab file etc.) in Python routines, to extract information from the input data and to visualize it. They will be able to use common statistics terminology and to implement it into Python programs by themselves or by using Python
packages (such as stasmodels)

Bibliography

1. Thomas Haslwanter, An introduction n t o Statistics with Python with applications in the life sciences, Springer (2016)
2. Phillip l. Good and James W. Hardin, Common errors in statistics an how to avoid them, Willey (2002)
3. José Unpingo , Python for probability, statistics, and machine learning, springer (2016)

Wireless Networks (Lynda zitoune) (E4 semester 2)

Content: Introduction to wireless communication networks; Examples of wireless networks: LoRa, SigFox,...; Transport protocols; Application: IoT development framework

Aim: This unit aims to give basic notions on wireless networks, especially those for IoT like: LoRa, Sigfox and Zigbee. Moreover, upper layers are presented to complete the IoT protocol stack as well as the associated development environment.

Optimization (Nabil Mustafa) (E4 semester 2)

Content: Linear programing; integer programs and mixed integer-linear programs; simplex algorithm; branch and bound and cutting plane techniques.

Aim: This course provides an introduction to linear programming (LP), which is one of the easier and more versatile tool capable of modeling both objective functions and constraints. We study modeling; the relationship between objectives and constraints; the simplex algorithm; duality and limit cases. In a second part, we study integer programming (IP), which is capable of modeling and solving most combinatorial problems, including NP-complete problems. This is essential in computer science because such problems are very common. Many examples are given. In a third part, we study transport problems, which is a class of IP problems that can be solved exactly and very efficiently. It is essential to be able to recognize such problems. In a last, short section, we provide an introduction to nonlinear, convex programming.

Bibliography Winston, W. L. Operations Research, Applications and algorithms, 3rd Ed. Duxbury Press, 1993.

Machine Learning (Yasmina Abdeddaïm) (E4 semester 2)

Content: Linear regression, Classification, Cross-validation, Model selection, Regularization, Support Vector Machines, Unsupervised learning

Aim: Statistical learning refers to understanding and handling data in order to learn a function *f* that models the relationship between predictor variables and the predicted variable(s). Mathematical tool will be used for that: probability and statistics (to handle uncertainty and randomness), linear algebra (to handle high-dimensional problems), optimization (to give the best possible answer). The algorithm's implementation is done with Python.

Bibliography :

1. *An Introduction to Statistical Learning'*, G.James, D.Witten, T.Hastie, R.Tibshirani, Springer, 2013; http://www-bcf.usc.edu/~gareth/ISL/code.html
2. *Statistical Machine Lerning'*, Kyoto U., M.Cuturi; http://marcocuturi.net/sml.html
3. *upyter notebook'* for implementation in Python by the mean of Anaconda navigator https://docs.anaconda.com/anaconda/navigator/

Model Checking (Yasmina Abdeddaïm) (E4 semester 2)

Content: Reactive Systems Modeling; Temporal Logics (CTL, LTL, CTL*); Model Checking Algorithms; NuSMV Model Checker.

Aim: This course is an introduction to model checking, an automatic verification technique of concurrent and reactive systems. First we introduce the Kripke structure as a model of reactive concurrent systems, then the linear (LTL) and branching time (CTL) temporal logics used to model temporal specifications. Finally we present LTL and CTL model checking algorithms and the model checker tool NuSMV.

Bibliography

1. Clarke, Orna Grumberg and Doron Peled, Model checking MIT Press Cambridge, MA, USA, 1999
2. NuSMV : http ://nusmv.fbk.eu/

Embedded Operating Systems (Yasmina Abdeddaïm) (E5 semester 1)

Content: The different tools used on the target system: kernel, uboot, rootfs, busybox ; Native compilation of a program on the target machine ; The tools used for Cross-compilation ; Develop/deploy an application to be used on the board ; Construction of a minimal image for a Raspberry Pi ; Customization of the image.

Aim: This course aims to present the methods and tools needed to build a GNU/Linux operating system using the source code. During the practical, students have to build their own Linux operating system embedded on a Raspberry-pi 2.

Bibliography https://buildroot.org/

Machine Learning 2 (Giovani Chierchia) (E5 semester 1)

Content: Supervised learning (reminder); Neural networks; Deep learning; Convolutional Neural Networks; Reinforcement. Learning

Aim: This course continues the study of machine learning concepts. We begin with a reminder of supervised learning then we move into the world of neural networks and deep learning. Each subject is illustrated in a laboratory session using simulated and real examples.

Bibliography : *An Introduction to Statistical Learning'*, G.James, D.Witten, T.Hastie, R.Tibshirani, Springer, 2013; http://www-bcf.usc.edu/~gareth/ISL/code.html

Image Analysis (Laurent Najman) (E5 semester 1)

Contents: Introduction to image processing; Image enhancement; Image transform; Image enhancement in the frequency domain; Optimization for image segmentation and filtering.

Aim: In this class you will learn the basic principles and tools used to process images, and how to apply them in solving some practical problems. Digital images are everywhere these days in thousands of scientific (e.g., astronomical, bio-medical), consumer, industrial, and artistic applications. Moreover they come in a wide range of the electromagnetic spectrum - from visible light and infrared to gamma rays and beyond. The ability to process images is therefore an incredibly important skill to master for engineering/science students, software developers, and practicing scientists. This course will cover the fundamentals of image processing.

Bibliography http ://www.imageprocessingplace.com/DIP-3E/dip3e main page.htm

ESIEE PARIS

Data Analysis for Developers (E5 semester 2)

Contents: Agile methodology; Data analysis; Machine Learning; Docker; Kubernetes.

Aim: This course will prepare you to work in a team on a data science project in a corporate environment. Using agile methodologies and the latest technologies, your team will deliver a web UI sitting on top of a microservice that returns predictions from a trained machine learning model from data your team has extracted from a relational database all in a containerized environment. You will learn how to read and write data into HIVE tables on Hadoop (HDFS), use HDFS to store and extract files, train machine learning models using your library of choice (MLLib, H2O.ai, Scikit-Learn, etc.), write distributed code using Spark, create and consume microservice APIs with frameworks like Python Django, and work with containers using Docker, Docker Compose, or Kubernetes.

Critical application Development (Yasmina Abdeddaïm) (E5 semester 1)

Contents: Synchronous paradigm; Esterel Synchronous Language; Lustre Synchronous Language; SCADE software.

Aim: Synchronous languages are used for the development of embedded reactive systems which are at the same time complex and critical systems. These languages are currently used successfully in industry, for example, EADS uses the software SCADE, based on the language Lustre, to program the embedded software flight control of Airbus. In this course, the fundamental principles of synchronous programming are presented then we present the language Lustre. Finally, we will learn the fundamental of the software SCADE and use it in a project.

Bibliography SCADE-SUITE: http://www.esterel-technologies.com/products/scade-suite/

Computer Vision (Xavier Hilaire) (E5 semester 1)

Contents: 2D and 3D projective transformations; Camera calibration; Epipolar geometry; Dense and spare stereo vision algorithms; Trifocal tensor.

Aim: This course introduces the fundamentals of computer vision: projective geometry, homographies, and single view cameras are first deeply studied to provide good mathematical grounds to attendees. Mathematical notions on eigenvalues, SVD, least squares, and constrained optimization will be recalled. Stereo vision and epipolar geometry are presented next, along with classical point matching algorithms and some efficient 3D object recognition algorithms. Extension to n cameras, and related open problems, are presented at last. The course includes a number of practical lab sessions on OpenCV, during which students will learn to: calibrate cameras, build a panoramic view from several overlapping shots, recognize and track a 3D object.

Distributed control Systems (Arben Cela) (E5 semester 1)

Content: Introduction to Distributed Control Applications; Modeling and analysis of Distributed Control System; Model Based Design of distributed real time application in Automotive Industry; Applications of Distributed Real Time Systems.

Aim: The main objective of this course is to give the necessary tools for modeling, analysis and design of Distributed control Systems. Classification of difference communication and calculation architectures help to achieve the modeling/analysis objectives and allows to point out the advantages/disadvantages w.r.t performance specification.